PCT/GB 98 / 03753

The Patent Office

15 DECEMBER 1998

INVESTOR IN PEOPLE

09/555929

GB98/03753

REC'D	08 FEB 1999
WIPO	PCT

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP9 1RH

5

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.

Signed

Dated 6/1/99.

PRIORITY DOCUMENT

SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

Patents Form 1/77

The Patent Office

Patents Act 1977
(Rule 16)

05JUN98 E365833-1 D03052
P01/7700 25.00 - 9812060.3

The Patent Office
Cardiff Road
Newport
Gwent NP9 1RH

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help you fill in this form)

1. Your reference

A25583

2. Patent application number
(The Patent Office will fill in this part)

0 4 JUN 1998

9812060.3

3. Full name, address and postcode of the or of each applicant *(underline all surnames)*

BRITISH TELECOMMUNICATIONS public limited company
81 NEWGATE STREET
LONDON, EC1A 7AJ, England
Registered in England: 1800000

Patents ADP number *(if you know it)*

1867002

If the applicant is a corporate body, give the country/state of its incorporation

UNITED KINGDOM

THE PATENT OFFICE
L
- 4 JUN 1998
LONDON

4. Title of the invention

DATA COMMUNICATIONS

5. Name of your agent *(if you have one)*

David WELLS

"Address for Service" in the United Kingdom to which all correspondence should be sent *(including the postcode)*

BT GROUP LEGAL SERVICES
INTELLECTUAL PROPERTY DEPARTMENT
HOLBORN CENTRE
120 HOLBORN
LONDON, EC1N 2TE

6939300001

Patents ADP number *(if you know it)*

1867001

6. If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and *(if you know it)* the or each application number

Country	Priority application number *(if you know it)*	Date of filing *(day / month / year)*

7. If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application

Number of earlier application	Date of filing *(day/month/year)*

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? *(Answer 'Yes' if:*

a) any applicant named in part 3 is not an inventor, or
b) there is an inventor who is not named as an applicant, or
c) any named applicant is a corporate body.

See note (d))

YES

9. Enter the number of sheets for any of the following items you are filing with this form. Do not count copies of the same document

Continuation sheets of this form	
Description	**15**
Claim(s)	**4**
Abstract	**1**
Drawing(s)	**11** +11

10. If you are also filing any of the following, state how may against each item

Priority Documents	
Translations of priority documents	
Statement of inventorship and right to grant of a patent *(Patents Form 7/77)*	
Request for preliminary examination and search *(Patents Form 9/77)*	
Request for substantive examination *(Patents Form 10/77)*	
Any other documents *(please specify)*	

11. I/We request the grant of a patent on the basis of this application.

Signature(s) D. Wells — Date: **4 JUNE 1998**

David WELLS, Authorised Signatory

12. Name and daytime telephone number of person to contact in the United Kingdom — **Rohini R Ranjitkumar** — **0171 492 8146**

Warning

After an application for a patent has been filed, the Comptroller of the Patent Office will consider whether publication or communication of the invention should be prohibited or restricted under Section 22 of the Patents Act 1977. You will be informed if it is necessary to prohibit or restrict your invention in this way. Furthermore, if you live in the United Kingdom, Section 23 of the Patents Act 1977 stops you from applying for a patent abroad without first getting written permission from the Patent Office unless an application has been filed at least 6 weeks beforehand in the United Kingdom for a patent for the same invention and either no direction prohibiting publication or communication has been given, or any such direction has been revoked.

Notes

a) If you need help to fill in this form or you have any questions, please contact the Patent Office on 0645 500505.

b) Write your answers in capital letters using black ink or you may type them.

c) If there is not enough space for all the relevant details on any part of this form, please continue on a separate sheet of paper and write "see continuation sheet" in the relevant part(s). Any continuation sheet should be attached to this form.

d) If you have answered 'Yes' Patents Form 7/77 will need to be filed.

e) Once you have filled in the form you must remember to sign and date it.

f) For details of the fee and ways to pay please contact the Patent Office.

DATA COMMUNICATIONS

The present invention relates to a data communications system, and in particular to the control of access by users to copies of digitally encoded data. It is applicable, for example, to the control of data multicast on the internet.

Multicasting routing techniques have been developed to allow multiple copies of a data item to be distributed efficiently to a large number of end users. However, for such techniques to be exploited commercially, it necessary to control selectively access by users to the data. For example, in an application in which selected stock market prices are multicast via the internet to subscribers, it is necessary to ensure that the data is accessed only by users who have paid the relevant subscription. This might be achieved by encrypting the data and only releasing the relevant key to the user in return for the subscription payment. However, whenever one user's subscription expires, after a fixed length of time or after a predetermined quantity of data has been received, it would be necessary to change the key for all the users, in order to exclude the one user. In such a situation, the traffic associated with key distribution becomes a significant operating overhead, and may even exceed the traffic for the data itself.

According to a first aspect of the present invention, there is provided a method of distributing digitially encoded data, comprising

dividing said data into a multiplicity of frames,

encrypting said frames,

distributing multiple copies of the said data frames to a multiplicity of users,

communicating a seed value for key generation to respective secure modules located at each of the multiplicity of users,

decoding the data frames at respective users using keys derived from the seed value communicated to the secure module,

passing a control message to the secure module at a selected one or more of the multiplicity of users,

at the or each selected user, in response to the said control message, controlling the availability of keys generated from the said seed value, thereby selectively controlling access by the users to the said data.

The method of the present invention provides full and effective control of access by users to data, without imposing heavy communication overheads. This is achieved by dividing the data item into frames, individually encrypting th frames with a series of keys, and using a controlled secure module at the customer location to generate the corresponding series of keys required to decrypt the received data. The secure module is controlled to limit the availability of the keys. For example, an intial set-up message to the secure module may instruct it only to generate a limited number of keys, say one hundred. If the user subsequently pays to extend their subscription, then a further control message may be sent to the secure module to allow the generation of further keys from the existing seed value

Although the invention is suitable for use in a multicast data communications network, it is also applicable in a wide range of other contexts, wherever it is necessary control access to a widely distributed data items. Possible applications include multicast audio/video streams for Video-on-Demand, network radio or surveillance; controlling access to the contents of CD ROMS or other storage media carrying software or multimedia data; controlling access to a set of vouchers giving access to other services; a multicast stream of messages such as stock prices, communications network prices, electricity prices, network management messages, news items, portfolio information, team briefings, standards documents, academic papers, magazines, product reviews, black lists, criminal intelligence, legal precedents, property profiles, pharmaceutical test results etc; a sequence of multicast messages within a network game, virtual world or simulation scenario (e.g. an aptitude exam), possibly just those messages that control access, but also possibly any data messages for which proof of reception is crucial to the fairness of the result of the simulation.

Control messages need not be distributed on-line. They may be distributed by any suitable means (e.g. on plastic cards, bar-codes, microdots, floppy disks etc.).

Preferably a control field is distributed to each of the multiplicity of users, and the secure module is arranged to enable decryption of a respective frame only when the said control field has been passed to the secure module. Preferably the said control message for modifying the availability of keys is communicated to the secure module in the said control field.

These preferred features of the invention, make it more difficult for the user to circumvent the control exercised through the key generation system. By passing a control control field to the secure module with each frame, and only allowing decryption when the control field is received they protect against interruption of the control channel to the secure module.

Preferably, each data frame includes a frame identity field, and each key generated by the secure module is specific to one frame identified by the said field.

As is further discussed below, the security of the system is further enhanced by including a frame identity field, and making the process of decryption dependent on the frame identity.

The method may include generating and storing a receipt for each frame decrypted by the user. The use of receipts generated in this manner is described and claimed in the present applicant's co-pending British Patent Application number 9726934.4, filed 19.12.97, Agent's reference A25546. The contents of that earlier application are incorporated herein by reference.

The user may receive and process the data frames using an appropriate terminal such as a personal computer or any other appropriate device, such as, for example, a Java-enabled mobile cellular telephone. The secure module provides a region in the customer terminal which is effectively under the control of the data provider, and which is not readily accessible to the customer. The secure module may simply be a software module which executes a cryptographic algorithm. This might be implemented, for example, as a Java program distributed by the operator of the remote data source as part of the process of setting up a session. To provide still higher levels of security, it is preferred that the secure module should include a dedicated processor and store located within a physically secure housing. Examples of such secure modules include smartcard structures, and cryptographic PC cards.

When the secure module has only a relatively low processing power, as may be the case, for example, when it is a smartcard, then preferably that module is required simply to output the different respective keys. Other processes running in the main part of the customer terminal are then responsible for decrypting the data frames. Alternatively, when the secure module has more processing power, as when, for example, a cryptographic co-processor card is used, then preferably the encrypted data frames are passed to the secure module and the module

generates the respective keys, decrypts the frames, and passes the decrypted frames out, for example, to an application program running on the customer terminal.

Preferably the control message is distributed with a data frame to the multiplicity of users, a user identity field identifying a selected user or group of users is included in the control message, and the control message is acted on only by the user or group of users identified by the said user identity field. The control message may include a stop flag and a contact sender flag. For example, the contact sender flag might be used to initiate a remote procedure call from the customer terminal to the data source, allowing a new key generation policy to be communicated to the terminal.

According to a second aspect of the present invention, there is provided a data communications system comprising

a) a remote data source arranged to output a plurality of frames;

b) encryption means for encrypting the plurality of frames with different respective keys;

c) a communications channel arranged to distribute multiple copies of the encrypted data frames ;

d) a multiplicity of customer terminals arranged to receive from the communications channel respective copies of the encrypted data frames;

e) a key generator located at a customer terminal and programmed to generate from a seed value keys for use in decrypting data frames:

f) key control means connected to the key generator, the key control means comprising:

an interface for receiving control messages; and

control means responsive to the said control messages and arranged to control the availabiltiy to the user of keys generated from the seed value; and

g) decryption means connected to the key generator and arranged to decrypt the data frames received at the customer terminal from the communications channel.

The invention also encompasses customer terminals and data servers.

Systems embodying the present invention will now be described in further detail, by way of example only, with reference to the accompanying drawings in which: Figure 1 is a schematic of a data communication system embodying the network;

Figure 2 is a schematic showing in further detail the functional components of the customer terminal in the system of Figure 1;

Figure 3 is a flow diagram showing the principal phases of operation of the system of Figure 1;

Figure 4 is a flow diagram showing in further detail the verification phase;

Figure 5 is a flow diagram showing in further detail the initialisation phase;

Figure 6 is a flow diagram showing in further detail the received/decrypt phase;

Figure 7 is a flow diagram showing in further detail the receipt phase;

Figure 8 shows the software architecture of the customer terminal;

Figure 9 shows the software architecture of the data server;

Figures 10a and 10b shows the structure of a data frame;

Figure 11 shows message flows in the data communications system.

As shown in Figure 1, a data communications system includes a data server 1 ("sender's machine") connected to a number of customer terminals 2 via a data communications network 3. Although for ease of illustration only a few customer terminals are shown, in practice the data server 1 may communicate simultaneously with many terminals. In the present example, the data communications network 3 is the public Internet. The sub-networks and the associated routers connecting the data server to the customer terminals support IP (Internet Protocol) multicasting.

In the present example, the data server 1 is a video server. The data server reads a video data stream from a mass storage device and compresses the data using an appropriate compression algorithm such as MPEG 2. An encryption module in the data server 1 then divides the compressed video data stream into frames. For example each frame may comprise data corresponding to one minute of the video signal. An encryption algorithm, such as that described in further detail below, then encrypts the frames of data. A common encryption algorithm is

used for all of the frames in one session. However, a sequence of keys is used, with a different key for each successive frame.

At each customer terminal, incoming data frames are processed using secure module 4. As described in further detail below, the secure module 4 generates a sequence of keys corresponding to those used originally to encrypt the data frames. The number of keys to be generated in a given session are determined by a contract between the user and the operator of the data server. For example, in the case of video-on-demand, the user might select program material, in response to which the server identifies the number of keys required to decrpt all the frames in the programme, and the cost of the programme. In return for payment from the user, the server sends the seed value for the key, together with a control instruction for the secure module to generate the required number, e.g. one hundred, of the keys. The keys may be passed out to the main processor of the customer terminal to allow the data to be decrypted. Alternatively, the secure module itself may carry out the step of decryption. In either case, the secure module stores a record of the keys generated. This record may comprise, for example, a count of the total number of keys issued in the course of a session, together with a session ID and a record of the time of the session.

During the course of the session, control signals may be sent to modify the access rights of the customer. For example, the user might choose to quit a program at an early stage and to gain a refund. This is effected by transmitting from the data server a data frame which contains, in addition to the data itself, a control message including the identity of the particular customer or group of customers whose access rights are to be modified. The control message may include a simple "stop" flag which, when set causes the secure module to cease releasing keys. Possible formats for the communication of control signals are discussed in further detail below with respect to Figures 10A and 10B. Conversely, the user might choose to view additional programme material, in which case a control message may be sent to the secure module to increase the number of keys to be generated e.g. from 100 to 200. Other changes in status are also possible. The frames may include a meta-data field which may be used to distinguish, for example, between different classes of subscriber. For example, subscribers might be divided into gold, silver and bronze classes, with gold users having access to data frames having meta-data values m1, m2 or m3, silver users having access to

m1 or m2, and bronze users having access to m1 only. In return for payment during the course of a session, the user might upgrade their subscription e.g. from bronze to silver, and thereby gain access to programme material carried in frames with m2 meta-data values in addition to material carried in frames with m1 meta-data values. The change is effected by the data server transmitting a control message to the secure module mandating key generation for m2 frames in addition to m1 frames.

Prior to commencing a session, a customer terminal 3 may have contracted with the operator of the data network 2 for a quality of service (QoS) which requires a specified minimum number of frames to be delivered per unit time. If subsequently, congestion in the network 2 causes the rate of frame delivery to fall below that specified in the contract, then the customer terminal 3 request from the data server 1 a refund of charges for the session. To validate this request, the data server 1 requests from the secure module 4 a "receipt". This receipt includes the data recorded in the data store and so provides a tamper-proof indication of the number of frames decrypted and made available to the customer in the course of a specified session.

Figure 2 shows the principal functional components of the customer terminal relevant to the present invention. A network interface 22 communicates data frames to and from the data network. The data frames pass from the interface 22 to a secure module 23. The secure module 23 has sub modules comprising a decryption module D a key generation module K and a secure store S. The key generation module passes a series of keys to the decryption module which decrypts a series of data frames received from the interface 22 and passes these to an application layer module 24. This carries out further processing and passes the resulting data to an output device, which in this example is a video display unit VDU 25. In a preferred implementation, the interface 22 may be embodied in hardware by an ISDN modem and in software by a TCP-IP stack. The secure module 23 may be, for example, a smartcard which is interfaced to the customer terminal via a PCMCIA socket The smartcard may use one of a number of standard data interfaces such as the Java card API (application programmer's interface) of Sun Microsystems, or the Microsoft smartcard architecture. Alternatively, the secure module may be embodied by a PCI cryptographic co-processor card such as that available commercially from IBM.

Figure 3 shows the main phases in the operation of the system described above. In phase P1, the server verifies that the secure module in the customer terminal is trustworthy and has a recognised identity. In phase P2 the secure module is initialised to decode data for a particular session. In phase P3 the data is transmitted and decryption carried out . During this phase a control message may be sent to the control module, for example to modify the number of frames which the user is allowed access to. In stage P4, which is optional, a receipt is generated. These phases will now be described in further detail.

When the secure module is, for example, a smartcard, then that smartcard is issued by the manufacturer with a unique public/private key pair. This key pair may be certified by a trusted third party. In phase P1, the server carries out steps to confirm that the smartcard does indeed come from a trusted supplier. The steps of phase P1 are shown in figure 4. In step S1 the server generates a random string. In step S2, the server sends the random string via the data network to the customer terminal. In step S3, the random data string is passed to the secure module (e.g. the smartcard). In step S4 the smartcard signs the random string. In step S5 the smartcard returns the signed string together with the relevant public key to the client application running on the customer terminal. In step S6, that client application returns the signed string and the public key via the data communications network to the server. In step S7 the server verifies the signed random string.

As shown in Figure 5, to set up the secure module to decode data in a particular session, the server first generates (s51) a seed value for use with an appropriate pseudo-random or chaotic function to generate a series of keys. It also generates a session key (s52). The server encrypts the seed value and a maximum number of keys to be generated using the secure module's public key (s3). It then transmits the encrypted seed value and maximum number of keys to be generated and the session key, to the customer terminal (s54). The client application passes the seed value and session key on to the secure module (s55). The secure module sets a packet counter to zero (s56) and initialises a sequence generator with the seed value (s57). The customer terminal is then ready to receive and decrypt data frames.

The server subsequently sends a series of frames to the client. Each frame has a frame number (also termed herein the packet number). Each frame

might also have a session key transmitted with it. The sequence of steps for the nth frame is illustrated in Figure 6. In step s61 the server sends the encrypted nth frame to the client. The client requests the key x for frame n from the secure module (s62). The secure module records the request (s63). The smartcard then returns the key x to the client (s64). The client deciphers the frame using x (s65). The client tests to determine with the frame is the last of a session (s66). If not then the steps are iterated for the n + 1th and subsequent frames.

In setting up the session, the customer has previously negotiated an agreement with the service provider as to the QoS level for the session. For an application such as video on demand this level may be stringent: for example the customer may require that no application-level frame is lost in transmission. If then this QoS level is not met, then the customer requests a refund from the service provider. The request for refund might specify, for example, that there was frame loss at a specified time into the video transmission. In processing such a request, the server requires a receipt from the customer. As shown in Figure 7, in step s71 the client requests a receipt for a specified session s from the secure module. The secure module reads the data which it recorded for that session and generates a receipt containing that data (s72). The secure module signs the receipt with the secure module's private key (s73). The secure module returns the signed receipt to the client (s74). The client in turn transmits the signed receipt to the server (s75). The server checks the signature on the receipt using the public key of the secure module (s76). The public key may be read from a database stored at the server. Having verified the signature, the server can then check the customers claim for a refund using the data contained in the receipt. This data may show, for example, a discrepancy between the number of frames decrypted in a session and the number transmitted by the server, thereby substantiating the customer's claim that a frame was lost.

The sequence used for generating the keys in the above example may be distributed to customers terminals using HTTP (hypertext transfer protocol) as Java code. A suitable chaotic function is:

$$x_{n+1} = 4rx_n(1-x_n)$$

When $r = 1$ this function takes and generates numbers in the range 0 to 1. A chaotic function such as this has the property that any errors in the value of x_n

grow exponentially as the function is iterated. In use, the secure module uses a higher accuracy internally than the accuracy of the key values exposed to the client. For example the secure module may use 128-bit numbers internally and then only return to the client the most significant 32 bits. In generating the key values, the chaotic function is iterated until the error in the value returned to the client grows bigger than the range. This then prevents the user guessing the sequence from the values returned by the secure module.

As an alternative or additional security measure, a different function may be used for each session. This serves to further reduce the possibility of the customer

Figure 10A shows the format of a frame transmitted in a first implementation of the system described above. The frame format is as follows:

1.Signature of Hash (2)

2.Hash of 3, 4, 5, 6

3.Key ID

4.Stop flag (y/n) (encrypted)

5.Contact sender flag (y/n) (encrypted)

6.Card IDs (encrypted)

7.Frame data

The frame is received at the network interface of a customer terminal and fields 1 to 6 are passed to the secure module. These comprise an encrypted block containing control fields as well as a key identity. This block is decrypted within the secure module . If the card ID is that of the secure module in question, then the secure module checks fields 4. and 5., the stop flag and contact sender flag. If the stop flag is set then no more keys are passed out. If the contact sender flag is set then the card does a remote procedure call to the sender (or the sender's representative) and gets a new key generation policy. The secure module then, unless instructed otherwise by the control fields, passes a key out for use in decrypting the frame data contained in field 7. The total length of the control fields passed to the secure module, and in particular the number of Card ID's (field 6), may be variable, in which case, in addition to the fields shown, a further,

unencrypted field is included before the control fields to indicate the total length of the control message. If the secure module does not receive a control field then it ceases to release keys. In this way neither accidental loss of a control message, nor intentional removal of such a message, can result in the customer gaining unauthorised access to data.

Figure 10B shows the format of a frame transmitted in a second implementation of the system described above. The frame format is as follows:

1.Signature of Hash (2) (signed with sender's public key)

2.Hash of 3, 4, and 5

3.Key ID

4.Control message (encrypted)

5.Card ID(s) (encrypted)

6.Frame data

The stack passes fields 1, 2, 3, 4 and 5 into the secure module to receive the key for 6. The use of this frame format relies upon a probablistic approach to controlling access. Every time a frame is sent it contains an encrypted control message and card ID which must be passed into the secure space along with the key ID to obtain the key. The control message may be a code representing the command "pass out no more keys". If the card receives this and the card ID(s) relate to it, it executes the command. If several users need to be excluded from a session, then their card IDs are rotated through different packets. In these examples the card ID constitutes the "user identity field" referred to in the claims below.

Figure 11 shows the message flows involved in setting up a session. Message 1 is a request from an application on the customer terminal for access, for example, to 100 frames of data. This message may be followed by other transactions (not shown) in the course of which the customer pays for the requested data, for example using a credit card number. Subsequently the sender transmits a set-up message , message 2 , to a secure module proxy on the customer machine. The control field from this set-up message is passed on to the secure module itself (message 3). The field may specify, for example, the number

of keys to be generated and for which frame numbers. It may also contain the seed value for key generation. An acknowledgement is then returned from the secure module to the proxy (message 4) from the proxy to the sender (message 5) and from the sender back to the application which generated the initial request (message 6). The interface between the sender and the proxy, indicated by the dashed ellipse, might be implemented, for example, using Java RMI (remote method invocation) or, as in this case, a CORBA interface.

Table 1 below list Java code for implementing a chaotic function. It returns the next number in a sequence, or the nth number in a sequence.

The key values need not necessarily be generated by a sequence. Instead other functions of the form k=f(seed, frame i.d.), where k is a key value, may be used. For example, the binary values of the frame identity might be used to select which of a pair of functions is used to operate on the seed value. Preferably a pair of computationally symmetric functions are used. For example , right or left-shifted XOR (exclusive OR) operations might be selected depending on whether a binary value is 1 or 0. If we label these functions A and B respectively, then, e.g., frame number six , i.e. 110, has a key generated by successive operations AAB on the seed value.

```
** Class to implement a chaotic sequence */

public class SecureSequence {


    protected int seqNum;

    protected double currNum;


    /** Create a SecureSequence object from a new seed */
```

```
    public SecureSequence (double currNum) {

      seqNum = 0;

      this.currNum = currNum;

    }



    /** Return the next number in the sequence */

    public int next() {

      ++seqNum;



      for (int i = 0; i < 20; ++i)  // 20 iterations is a guess,
could use less

        currNum = 4 * currNum * (1 - currNum);



      // return the most significant 32 bits of a 64 bit number



      return (int)((double)Integer.MAX_VALUE * currNum);
```

```
    }

    /** Return the current sequence number of the last number
returned */

    public int sequenceNumber() {
      return seqNum;
    }

    /** Return the number in the sequence at the requested
position in
        the sequence */

    public int next(int seqNum) {

      // if the number is too small return zero (should really be
an exception)
      if (seqNum <= this.seqNum) return 0;

      // iterate through the sequence to get to the right number
      while (this.seqNum != seqNum)
```

```
            int value=next();
        return value;

    }

}
```

CLAIMS

1. A method of distributing digitially encoded data, comprising

a) dividing said data into a multiplicity of frames,

b) encrypting said frames,

c) distributing multiple copies of the said data frames to a multiplicity of users,

d) communicating a seed value for key generation to respective secure modules located at each of the multiplicity of users,

e) decoding the data frames at respective users using keys derived from the seed value communicated to the secure module,

f) passing a control message to the secure module at a selected one or more of the multiplicity of users,

g) at the or each selected user, in response to the said control message, controlling the availability of keys generated from the said seed value, thereby controlling access by the users to the said data.

2. A method according to claim 1, in which a control field is distributed to each of the multiplicity of users, and the secure module is arranged to enable decryption of a respective frame only when the said control field has been passed to the secure module.

3. A method according to claim 2, in which the said control message for modifying the availability of keys is communicated to the secure module in the said control field.

4. A method according to any one of the preceding claims, in which each data frame includes a frame identity field, and each key generated by the secure module is specific to one frame identified by the said field.

5. A method according to any one of the preceding claims, in which the step of distributing multiple copies of the said data comprises multicasting packets of data via a communications network to the plurality of users.

6. A method according to any one of the preceding claims, in which the control message is distributed with a data frame to the multiplicity of users, a user identity field identifying a selected user or group of users is included in the control message, and the control message is acted on only by the user or group of users identified by the said user identity field.

7. A method according to any one of the preceding claims, in which the control message includes a stop flag, and in response to the stop flag the generation of keys at the or each selected user is stopped.

8. A method according to any one of the preceding claims, including returning a response signal from the secure module to the source of the control message.

9. A method according to claim 8, in which the control message includes a contact sender flag, and the step of returning a response signal from the secure module is carried out when the contact sender flag is set.

10. A method according to claim 8 or 9, including transmitting a further control message to the user on receipt of the said response signal.

11. A method of operating a customer terminal in a data communications system, the method comprising:

a) receiving at the customer terminal a multiplicity of encrypted data frames

b) receiving at the customer terminal a seed value for key generation

c) passing the said seed value for key generation to a secure module located at the customer terminal

d) generating in the secure module using the seed value keys for the decryption of data frames;

e) decrypting the data frames using the said keys;

f) passing to the said secure module a control message received from a source remote from the customer terminal;

g) in response to the said control message controlling the availability of keys generated using the said seed value and thereby controlling access by the user of the customer terminal to data received at the customer terminal.

12. A data communications system comprising

a) a remote data source arranged to output a plurality of frames;

b) encryption means for encrypting the plurality of frames with different respective keys;

c) a communications channel arranged to distribute multiple copies of the encrypted data frames ;

d) a multiplicity of customer terminals arranged to receive from the communications channel respective copies of the encrypted data frames;

e) a key generator located at a customer terminal and programmed to generate from a seed value keys for use in decrypting data frames:

f) key control means connected to the key generator, the key control means comprising:

an interface for receiving control messages; and

control means responsive to the said control messages and arranged to control the availabiltiy to the user of keys generated from the seed value; and

g) decryption means connected to the key generator and arranged to decrypt the data frames received at the customer terminal from the communications channel.

13. A data communications system according to claim 12, in which the communications channel is a packet-switched data network.

14. A customer terminal for use in a method according to any one of claims 1 to 11, the customer terminal comprising:

a) a data interface for connection to a data communications channel;

b) a key generator programmed to generate from a seed value keys for use in decrypting data frames:

c) key control means connected to the key generator, the key control means comprising:

an interface for receiving control messages; and

control means responsive to the said control messages and arranged to control the availabiltiy to the user of keys generated from the seed value; and

d) decryption means connected to the data interface and to the key generator and arranged to decrypt data frames received via the data interface.

15. A data server for use in method according to any one of claims 1 to 10, the data server comprising:

a) a data interface for connection to a data communications channel:

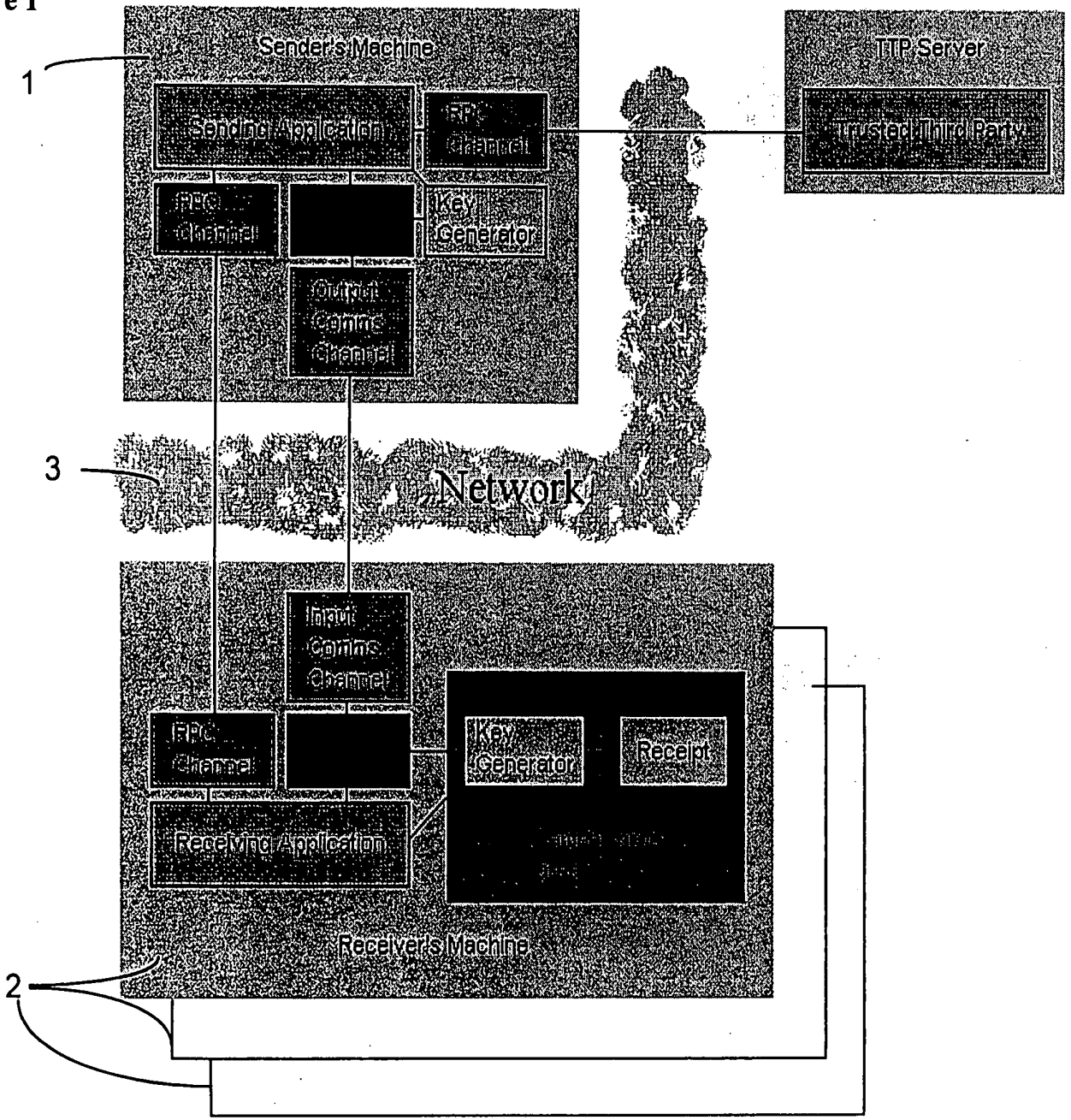
b) means for outputting encrypted data frames via the data interface onto the communications channel for receipt by a multiplicity of customer terminals;

c) means for outputting control mesages onto a data communications channel for controlling the operation of key generators at customer terminals.

ABSTRACT

In a data communications system, data is divided into a number of frames that are encrypted. Multiple copies of the frames are distributed to users. A seed value the generation of keys is also distributed. A secure module at each user generates keys for use in decoding the data frames. Control messages are passed to the secure module to control the generation of the keys, and hence to control the access by a selected user to the data.

Figure 1


1
Sender's Machine
Sending Application
RPC Channel
RPC Channel
Key Generator
Output Comms Channel
TTP Server
Trusted Third Party
3
Network
Input Comms Channel
RPC Channel
Key Generator
Receipt
Receiving Application
Receiver's Machine
2

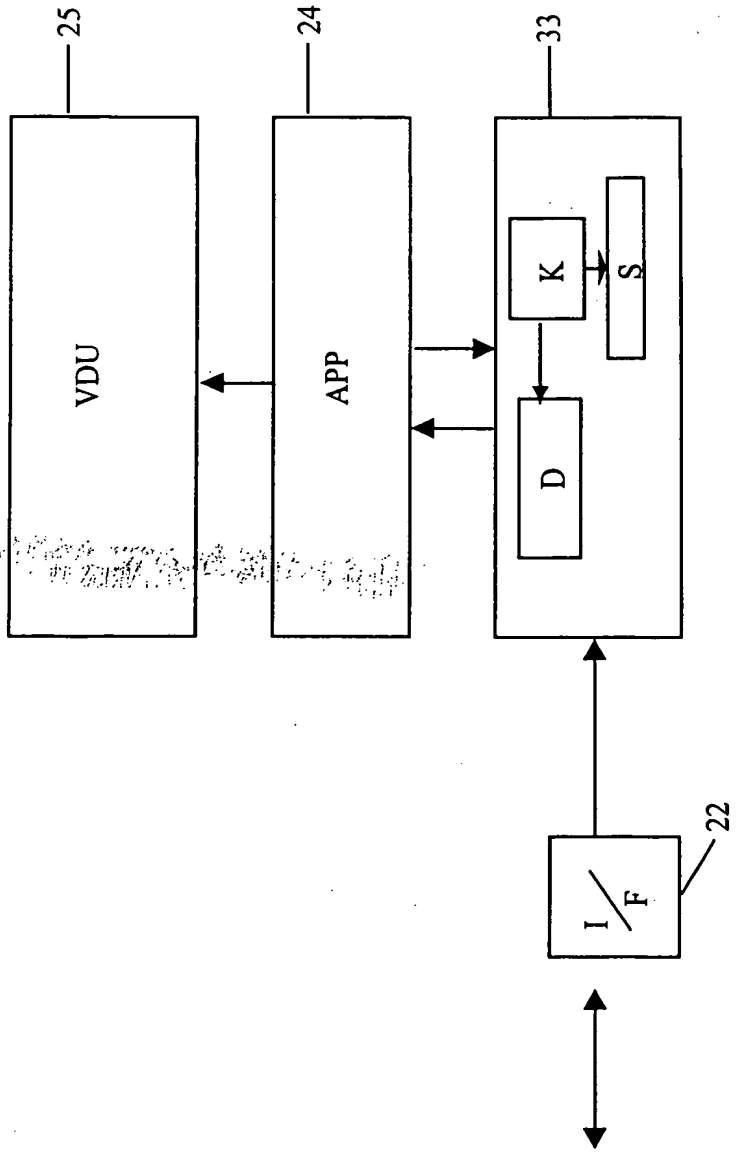
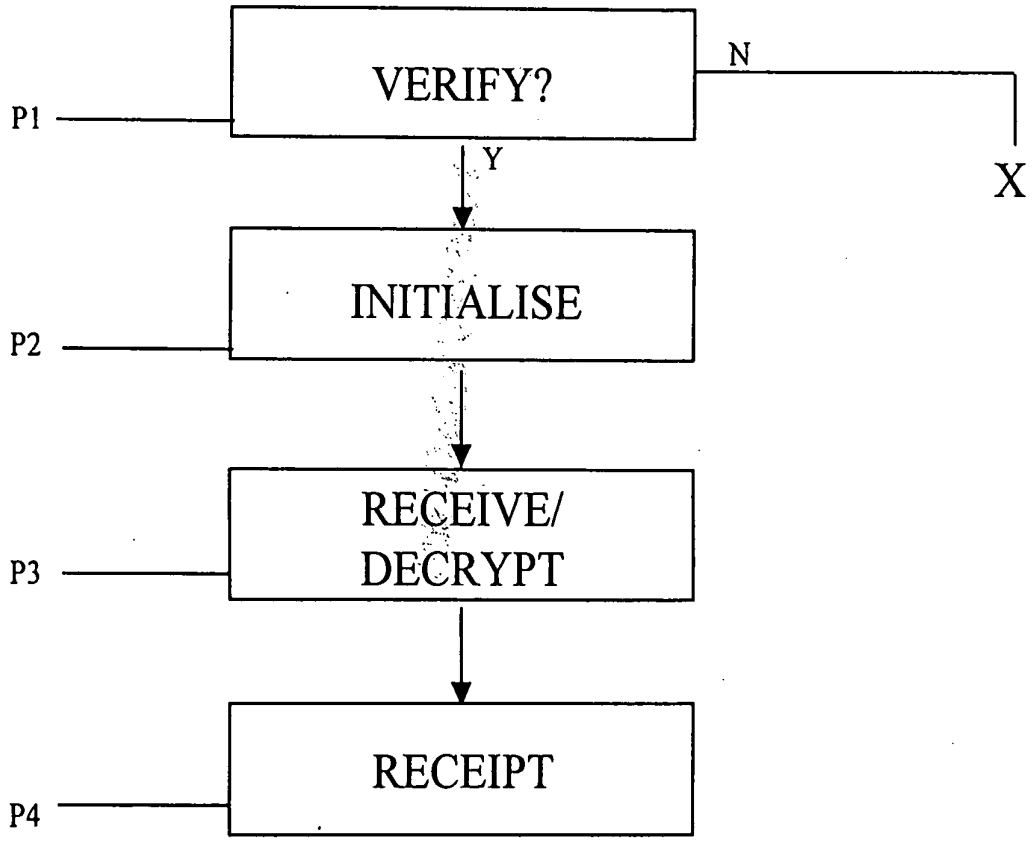


Figure 2

Figure 3



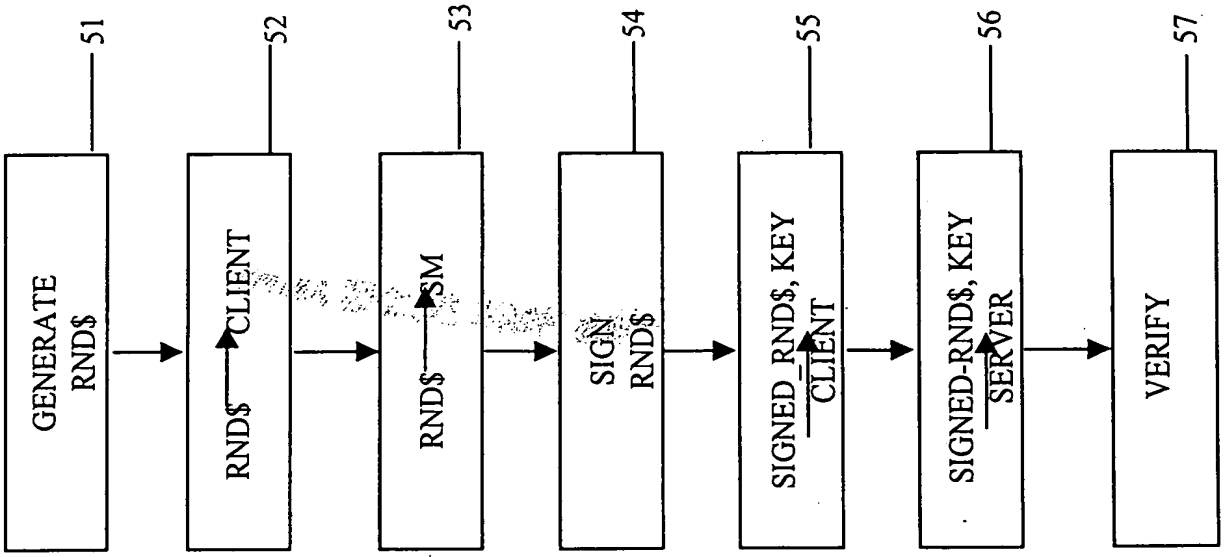

51
GENERATE
RND$
52
RND$
CLIENT
53
RND$
SM
54
SIGN
RND$
55
SIGNED_RND$, KEY
CLIENT
56
SIGNED-RND$, KEY
SERVER
57
VERIFY

Figure 4

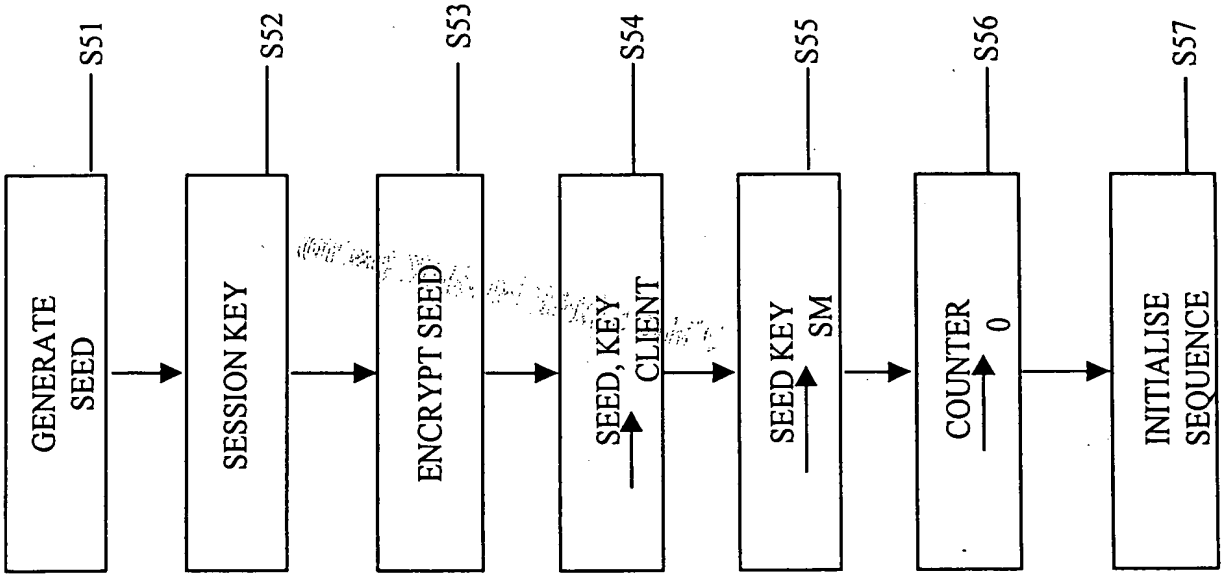

S51
GENERATE SEED
S52
SESSION KEY
S53
ENCRYPT SEED
S54
SEED, KEY CLIENT
S55
SEED KEY SM
S56
COUNTER 0
S57
INITIALISE SEQUENCE

Figure 5

Figure 6

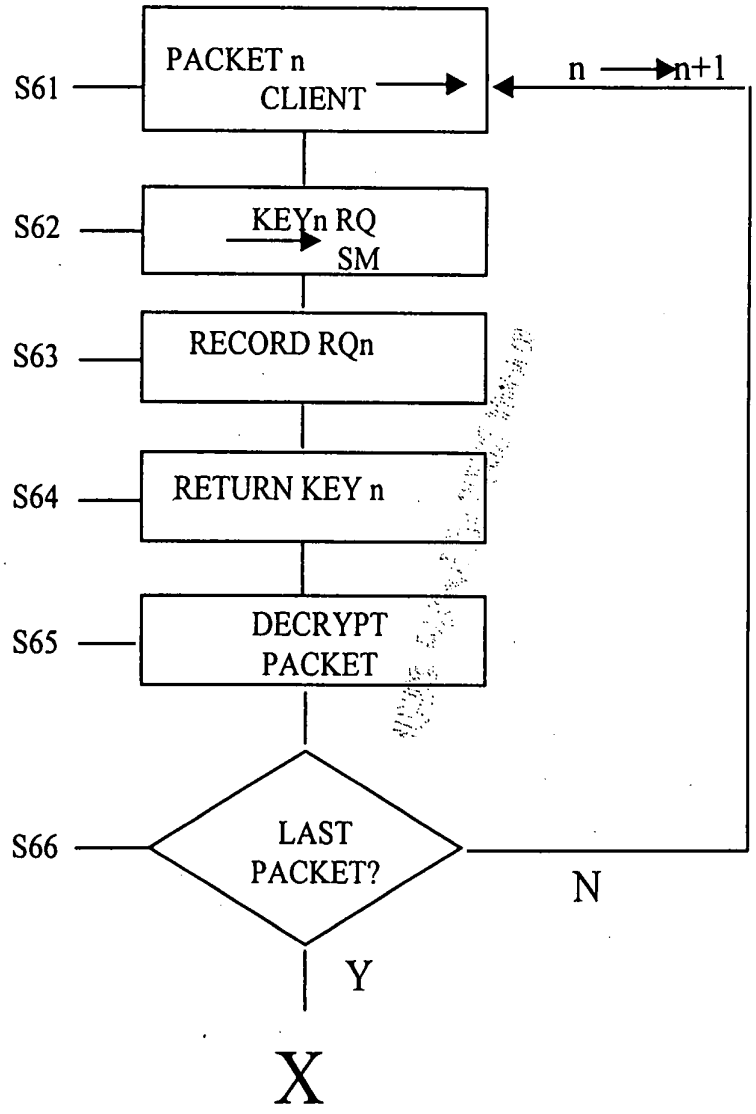

S61
PACKET n
CLIENT
n
n+1
S62
KEYn RQ
SM
S63
RECORD RQn
S64
RETURN KEY n
S65
DECRYPT
PACKET
S66
LAST
PACKET?
N
Y
X

Figure 7

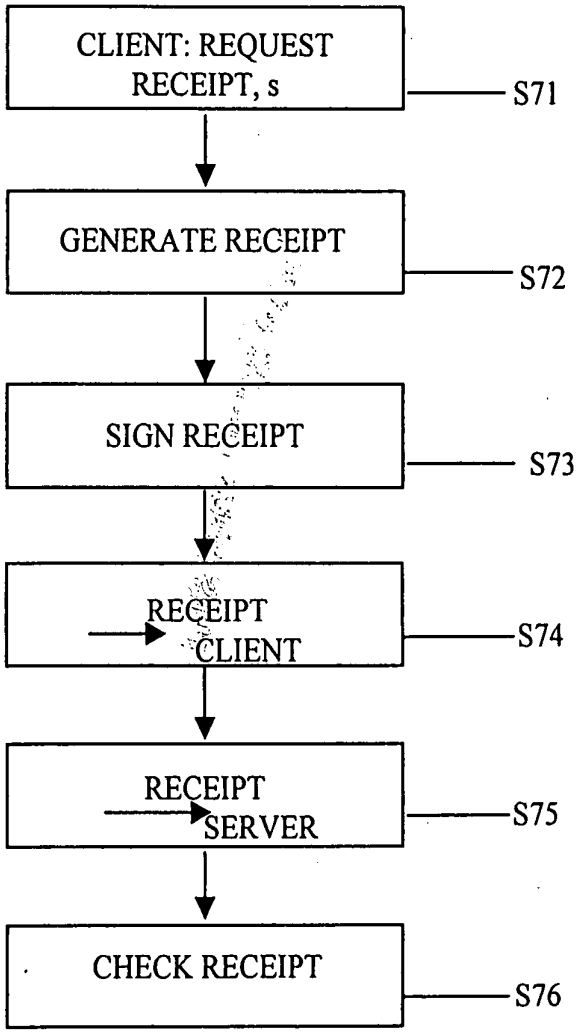
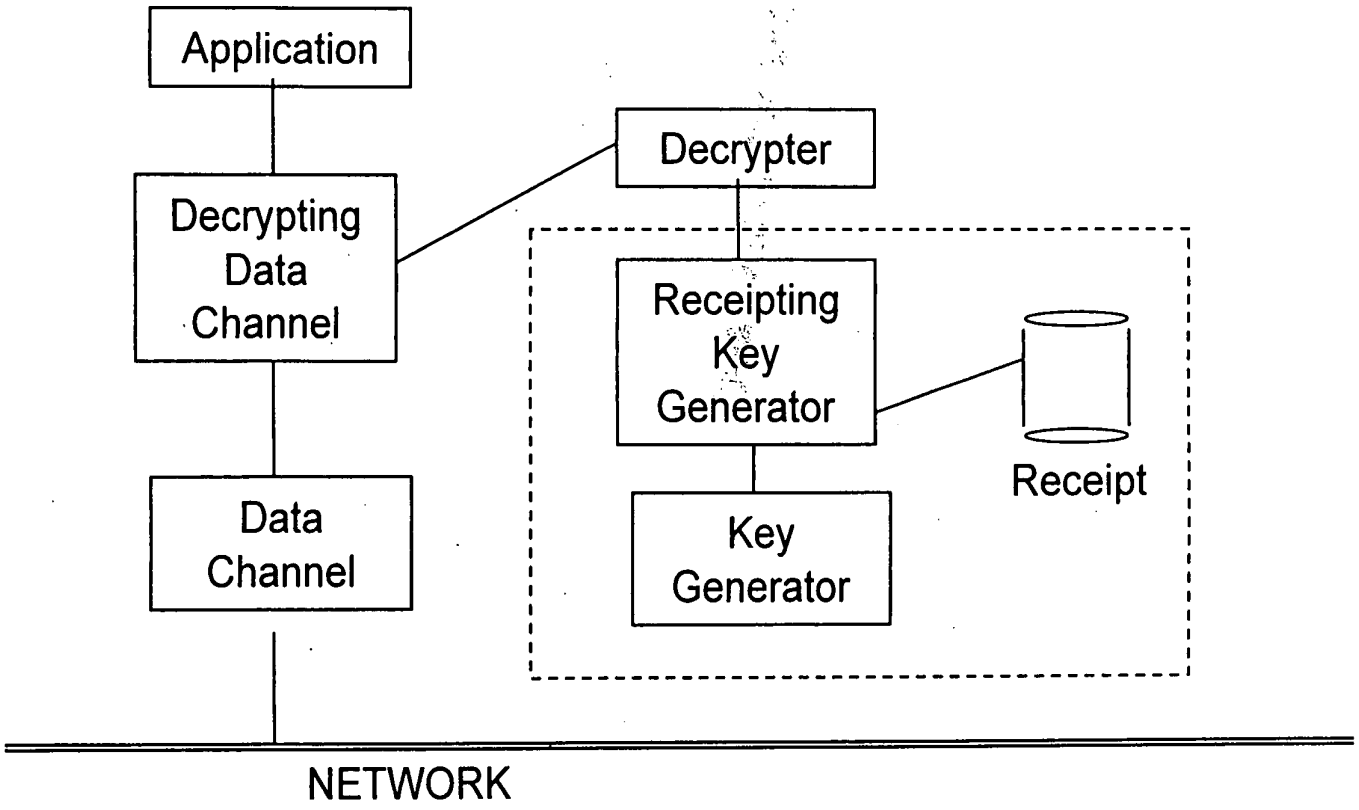

CLIENT: REQUEST RECEIPT, s
S71
GENERATE RECEIPT
S72
SIGN RECEIPT
S73
RECEIPT CLIENT
S74
RECEIPT SERVER
S75
CHECK RECEIPT
S76

Figure 8


Application
Decrypting Data Channel
Data Channel
Decrypter
Receipting Key Generator
Key Generator
Receipt
NETWORK

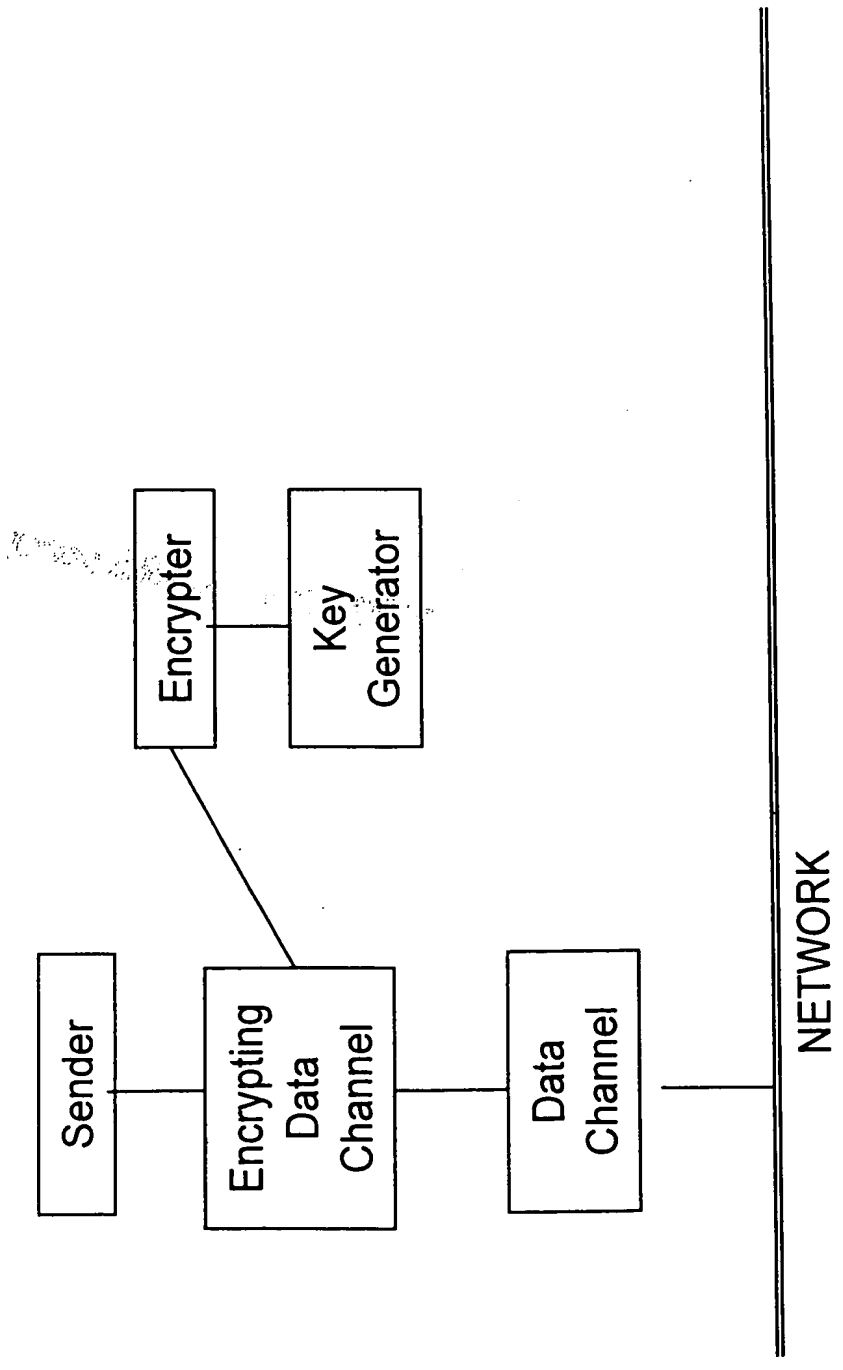

Encrypter
Key Generator
Sender
Encrypting Data Channel
Data Channel
NETWORK

Figure 9

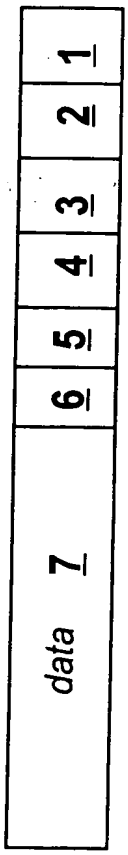


Figure 10A

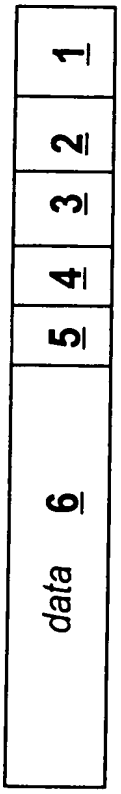
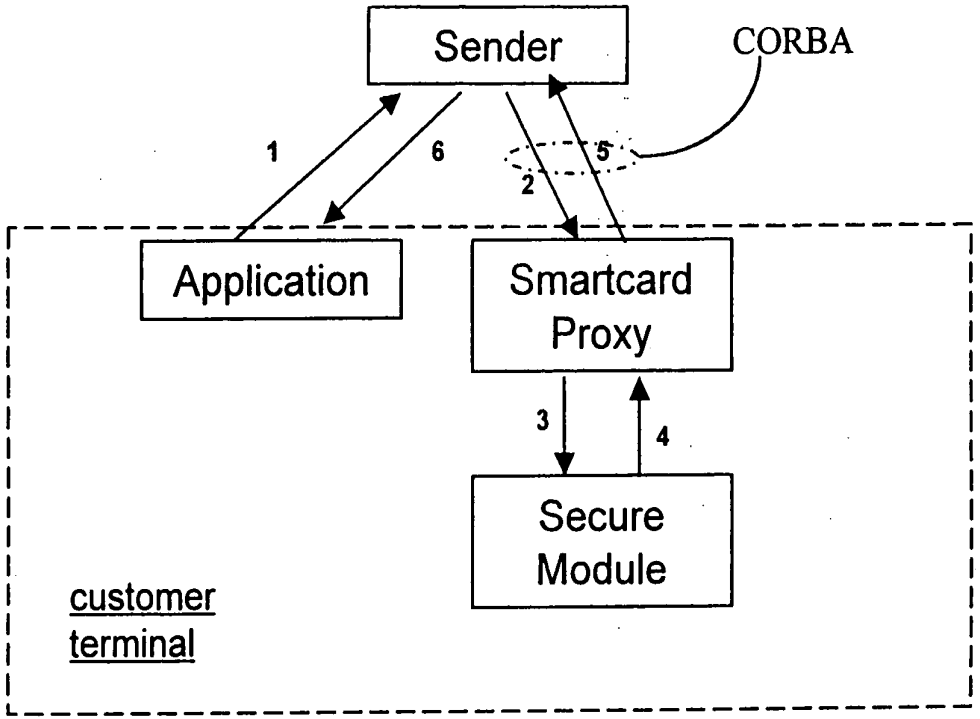


Figure 10B

Figure 11


Sender
CORBA
1
6
2
5
Application
Smartcard Proxy
3
4
Secure Module
customer terminal

PCT/GB98/03753

BT GROUP LEGAL SERVICES

15/12/98